

# 修士論文概要書

2013 年 2 月提出

学籍番号 5112B018-9

専門分野	情報理工学専攻	氏 名	川村 一志	指 導 教 員	戸川 望	印
研究指導	情報システム設計研究					
研 究 題 目	RDR アーキテクチャを対象とした熱考慮高位合成手法に関する研究					

## 1 はじめに

半導体の微細化技術向上に伴い、IC チップ内部の発熱、特にホットスポットと呼ばれる局所的に温度の高い空間が問題となっている。一方、微細化に伴ってゲート遅延より配線遅延が支配的となったため、高位合成段階で配線遅延を考慮する必要性が生じている。これら双方の問題に対処するため、本論文では配線遅延を考慮した設計が可能な RDR アーキテクチャを対象に、熱考慮高位合成手法を提案する。RDR アーキテクチャはチップ内部を同じ面積の島に分割するため、提案手法では演算の実行回数に注目して島間の消費電力量を均一化し、ホットスポットの温度を削減する。そして、面積の利用効率が悪いという RDR アーキテクチャの欠点を逆手に取り、空き領域に新たな演算器を追加することで温度の削減効果を高める。さらに、レジスタがチップ内部の発熱に与える影響を見積もることができるような評価関数、合成フローを改良する。計算機実験により、提案手法は従来手法と比較して最大 15.51% ホットスポットの温度を削減できることを確認した。

## 2 RDR アーキテクチャを対象とした熱考慮高位合成手法

提案手法が対象とする RDR アーキテクチャは、図 1 のようにチップ内が 2 次元配列の島に分割された構成を持つ。RDR アーキテクチャを対象とした熱考慮高位合成問題とは、DFG、RDR アーキテクチャの島数、島の容量制約、演算器ライブラリ、クロック周期制約が与えられたとき、RDR アーキテクチャ内の最大温度を最小化するように DFG をスケジューリング・バインディングし、演算器を各島に割り当てることである。

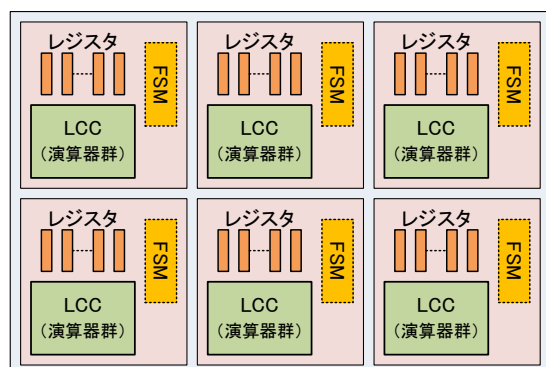


図 1: 3 × 2 の RDR アーキテクチャ。

### 2.1 演算の再割り当てと演算器の再配置による熱考慮高位合成手法

RDR アーキテクチャを対象とした演算の再割り当てと演算器の再配置による熱考慮高位合成手法の合成フローを図 2 に提案する。フローへの主な入力図 3 のような DFG と演算器の初期配置である。

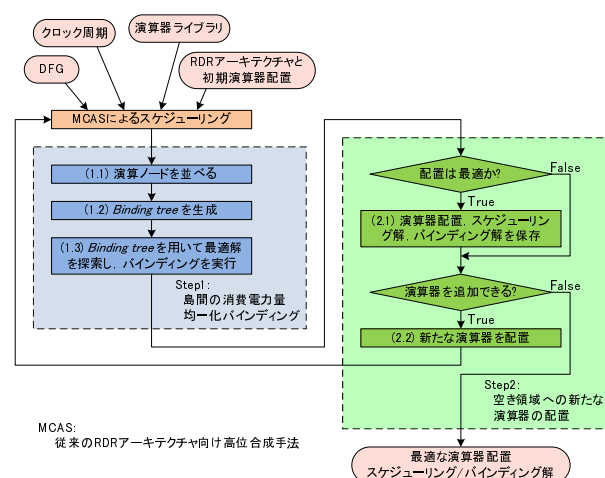


図 2: 合成フロー。

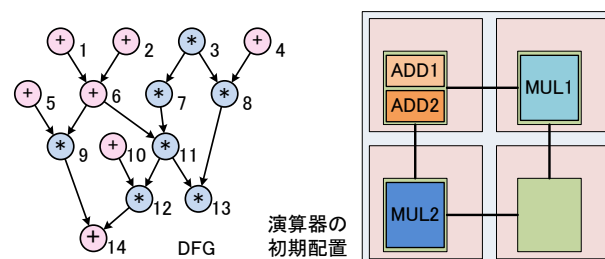


図 3: 合成フローへの入力。

DFG と演算器の初期配置を入力すると、MCAS によるスケジューリングが実行され、各演算の実行タイミングが決定される。続いてスケジューリング結果をもとに、島間の消費電力量を均一化するようにバインディングする。演算器を追加可能な島が存在する場合、演算器を配置して再度スケジューリング・バインディングを実行する。演算器の追加とスケジューリング・バインディングの反復は、すべての演算器追加パターンを評価し終わると終了となり、図 4 に示されるような反復中で最適と評価された場合の演算器配置およびスケジューリング/バインディング解を出力する。

提案手法はバインディング時および新たな演算器配置時の評価において、島内消費電力量を見積もる必要

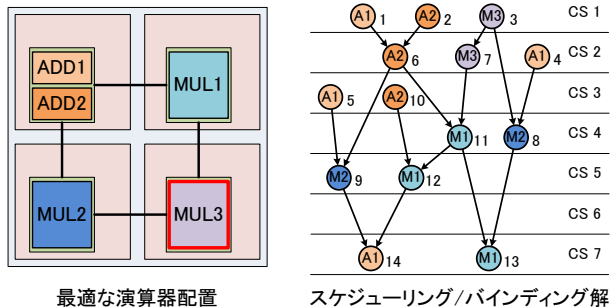


図 4: 合成フローからの出力。

がある。しかし、本提案手法では島毎に割り当てられたレジスタの影響を考慮することができず、島内消費電力量の見積もり精度が低いという問題がある。

## 2.2 島内消費電力量見積もりにもとづく熱考慮高位合成手法

RDR アーキテクチャを対象とした島内消費電力量見積もりにもとづく熱考慮高位合成手法の合成フローを図 5 に提案する。本提案手法は、前提案手法の問題点を解消した改良手法である。

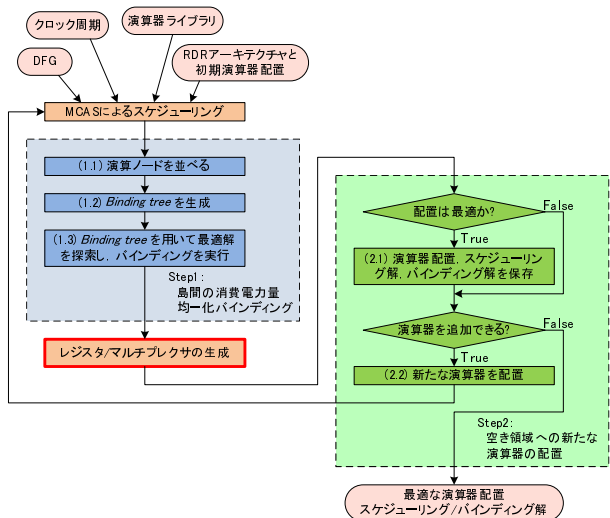


図 5: 合成フロー。

レジスタの各島への割り当てはスケジューリング/バインディング結果に依存するため、バインディング時の評価においてレジスタの影響を考慮して島内消費電力量を見積もることはできない。一方、新たな演算器配置時の評価ではスケジューリング/バインディングを完了しているため、図 5 のように合成フローを変更することでレジスタの影響を考慮して島内消費電力量を見積もることが可能である。この改良により、提案手法の効果を高めることができると考えられる。

## 3 計算機実験結果

各手法 (MCAS および 2 種類の提案手法) を C++ を用いて計算機上に実装した。HotSpot-5.0 を用いて

チップ内部とチップ表面の温度差をシミュレーションした結果は表 1 の通りである。

表 1: 実験結果。

App. (島数)	適用 手法	演算器数 (ADD, MUL)	最大温度 差 (°C)	削減割合 (%)
DCT (3 × 2)	MCAS	(4, 3)	8.80	—
	Ours 1	(6, 3)	8.33	5.34
	Ours 2	(4, 4)	7.53	14.43
EWF (2 × 2)	MCAS	(2, 2)	8.51	—
	Ours 1	(4, 2)	7.27	14.57
	Ours 2	(4, 2)	7.19	15.51
FIR (3 × 2)	MCAS	(3, 3)	8.07	—
	Ours 1	(4, 4)	7.80	3.35
	Ours 2	(3, 4)	7.47	7.43

図 6 は、アプリケーション DCT におけるチップ内部の温度分布を表す。提案手法を用いることで効果的にホットスポットの温度を削減することができる。

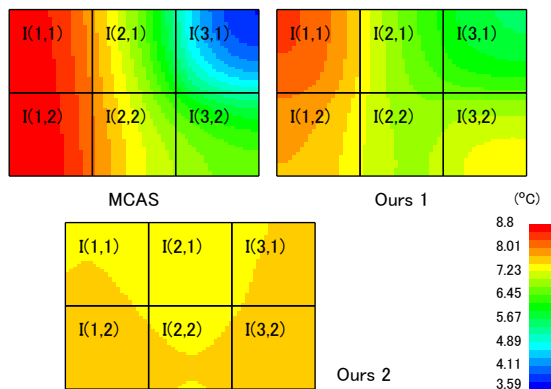


図 6: チップ内部の温度分布 (App.: DCT)。

## 研究業績

### 原著論文

1. K. Kawamura, et al., *IEICE Transactions on Fundamentals*, vol. E96-A, no. 1, pp. 312–321, 2013.

### 国際学会 (査読あり)

1. K. Kawamura, et al., *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, May. 2013. (accepted)

### 国内学会 (査読あり)

1. 川村一志, et al., 情報処理学会 DA シンポジウム 2012 論文集, vol. 2012, no. 5, pp. 133–138, Aug. 2012.

### 国内学会 (査読あり)

1. 川村一志, et al., 信学技報, vol. 112, no. 320, pp. 13–18, Nov. 2012.
2. 川村一志, et al., デザインガイア 2012, ポスター発表.

### 受賞

1. 情報処理学会 SLDM 優秀発表学生賞, Nov. 2012.